

SPECIFICATION

TO ALL WHOM IT MAY CONCERN:

BE IT KNOWN THAT WE, TAKU KODAMA, a citizen of Japan residing at Kanagawa, Japan, YASUYUKI NOMIZU, a citizen of Japan residing at Kanagawa, Japan, JUNICHI HARA, a citizen of Japan residing at Kanagawa, Japan, TOSHIO MIYAZAWA, a citizen of Japan residing at Kanagawa, Japan, HIROYUKI SAKUYAMA, a citizen of Japan residing at Tokyo, Japan, YASUYUKI SHINKAI, a citizen of Japan residing at Kanagawa, Japan, NEKKA MATSUURA, a citizen of Japan residing at Kanagawa, Japan, TAKANORI YANO, a citizen of Japan residing at Kanagawa, Japan and TAKAYUKI NISHIMURA, a citizen of Japan residing at Tottori, Japan have invented certain new and useful improvements in

IMAGE PROCESSING APPARATUS FOR COMPOSITING IMAGES

of which the following is a specification:-

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention generally relates to an image processing apparatus and an image forming apparatus, and more particularly, to an image processing apparatus that forms an image in which a plurality of component images are composited (formed into a composite image), an image forming apparatus in which the image processing apparatus is provided, a computer program that causes a computer to function as the image processing apparatus, and a computer readable recording medium storing the computer program.

2. Description of the Related Art

JPEG is one of the most widely used image compression/decompression algorithms. JPEG 2000, a successive technique to JPEG, is a further improved algorithm that offers higher performance and various additional functions. JPEG 2000 is expected to become the de facto standard of the next generation that realizes superfine compression/decompression of still pictures.

Recent improvements in the precision of image forming apparatuses have enabled one to composite a plurality of pages into fewer pages. The

rapid growth in information available through computers as well as concerns about environmental problems increase demand for such compositing of pages.

5 Conventionally, each image is decompressed, expanded in a memory, and shrunk to a certain size, if necessary.

 The conventional technique of compositing images, however, requires much more time and memory
10 space than processing of single images does. Users need to prepare more memory capacity and spend more time to process the compositing of images than to process the images as they are.

 To solve such problems, Japanese Patent
15 Laid-open Application No. 2000-156829 discloses an invention to save memory capacity by processing the compositing of images by lines or alternative small units.

 Japanese Patent Laid-open Application No.
20 2000-156830 discloses an invention in which an additional buffer is provided so as to accelerate the compositing of images.

 Additionally, Japanese Patent Laid-open
Application No. 2001-148774 discloses an invention in
25 which users can designate the number of images to be

composited.

Japanese Patent Laid-open Application No. 10-322542 discloses an invention for the compositing of previously composited images.

5 The inventions disclosed in the above first and second references, however, cannot solve the problem that each image to be composited is expanded in the memory one by one, taking much more time than processing of single images.

10 Neither of the inventions disclosed in the above third and fourth references is the solution to the problem.

SUMMARY OF THE INVENTION

15 Accordingly, it is a general object of the present invention to provide a novel and useful image processing apparatus with which at least one of the above problems is solved.

20 More particularly, it is an object of the present invention to composite a plurality of component images and to decomposite a composite image easily without expanding all the component images in the memory.

25 Yet another object of the present invention is to reduce the memory size of the composite images,

which results in saving memory capacity required for storing the composite images.

To achieve one or more of the above objects, an image processing apparatus, according to an aspect of the present invention, for compositing a plurality of component images into a composite image, includes: a separating unit that separates a codestream corresponding to each one of the plurality of component images into a header portion and a data portion, wherein said codestream is generated by dividing the corresponding one of the plurality of component images into one or more rectangular regions, transforming pixel values of the divided one or more rectangular regions with discrete wavelet transform into transform coefficients, and compressing said transform coefficients; a header processing unit that edits said separated header portion so as to compose the plurality of component images into said composite image; and a codestream generation unit that generates a codestream corresponding to said composite image by combining the edited headers and the separated data portions.

Accordingly, the plurality of component images can be composited into a composite image without repeatedly expanding each component image in

the memory.

Additionally, only a portion of code data that is needed for the composite image is retained, and the remaining portion of the code data that is
5 not needed for the composite image is abandoned. Accordingly, the file size of the composite image can be reduced, and the memory capacity required for storing the composite image can be reduced.

The image processing apparatus, according
10 to another aspect of the present invention, further includes a composite designating unit that designates a number of component images to be composited in said composite image; wherein said header processing unit edits said separated header portion based on the
15 designated number of component images.

Accordingly, the number of component images to be composited can be designated. A user can designate the number of component images that the user desires to composite.

20 The image processing apparatus according to yet another aspect of the present invention is characterized in that said composite designating unit designates at least one of the number of component images in horizontal directions and at least one of
25 the number of component images in vertical directions.

Accordingly, the number of component images to be composited can be designated by designating the number in the horizontal directions and/or the number in the vertical directions.

5 The image processing apparatus according to yet another aspect of the present invention further includes a code data processing unit that extracts predetermined code data of said separated data portion.

10 Accordingly, only a portion of code data that is needed for the composite image is retained, and the remaining portion of the code data that is not needed for the composite image is abandoned. Accordingly, the data amount of the composite image
15 can be reduced, and the memory capacity required for storing the composite image can be reduced.

 An image processing apparatus according to yet another aspect of the present invention, for decompositing (separating) a composite image into a
20 plurality of component images, includes: a separating unit that separates a codestream corresponding to said composite image into header portions and data portions, wherein said codestream is generated by dividing the corresponding composite image into one
25 or more rectangular regions, transforming pixel

values of the divided one or more rectangular regions
with discrete wavelet transform into transform
coefficients, and compressing said transform
coefficients; a header processing unit that edits
5 said separated header portions so as to generate a
plurality of new codestreams corresponding to said
component images; and a codestream generation unit
that generates a plurality of new codestreams
corresponding to said component images by combining
10 the edited headers and the separated data portions.

Accordingly, the image processing apparatus
can easily decomposite a composite image without
expanding the composite image in the memory.

An image forming apparatus according to yet
15 another aspect of the present invention includes: an
image input unit that reads an image of a document;
an image compression coder unit that generates a
single codestream by dividing the read image into one
or more rectangular regions, transforming pixel
20 values of the divided one or more rectangular regions
with discrete wavelet transform into transform
coefficients, and compressing said transform
coefficients; an image processing apparatus as
claimed in claim 1; and a printer engine that forms
25 an image on a recording medium based on each

codestream generated by said codestream generation unit of said image processing apparatus.

Accordingly, the image forming apparatus can easily composite a plurality of component images
5 into a composite image without repeatedly expanding each component image in the memory.

Other objects, features, and advantages of the present invention will become more apparent from the following detailed description when read in
10 conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram for explaining the JPEG algorithm;

15 FIG. 2 is a block diagram for explaining the JPEG 2000 algorithm;

FIGs. 3A - 3D are schematic diagrams showing sub-bands in the case that the decomposition level is three;

20 FIG. 4 is a schematic diagram showing components of a color image divided into tiles;

FIG. 5 is a schematic diagram for explaining the relationship between precincts and code-blocks;

25 FIG. 6 is a schematic diagram showing steps

of dividing transform coefficient by a bit-plane and ordering bit-planes by pixels or code-blocks;

FIG. 7 is a data diagram showing the structure of a codestream;

5 FIG. 8 is a cross-section of an image forming apparatus according to an embodiment of the present invention;

FIG. 9 is a block diagram showing the hardware structure of a controller of the image forming apparatus shown in FIG. 8;

10 FIG. 10 is a block diagram showing the functional structure of the image processing apparatus according to an embodiment of the present invention;

15 FIG. 11 is a block diagram showing the functional structure of an image compositing unit according to an embodiment of the present invention;

FIG. 12 is a data diagram for explaining the structure of a plurality of codestreams before compositing;

20 FIG. 13 is a data diagram for explaining the structure of a codestream after compositing;

FIG. 14 is a flow diagram for explaining an exemplary operation of the image forming apparatus for compositing a plurality of component images; and

25

FIG. 15 is a flow diagram for explaining an exemplary operation of the image forming apparatus for decompositing a composite image.

5 DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

 [JPEG 2000 Algorithm]

 The algorithm of JPEG 2000 related to the preferred embodiments of the present invention is described below.

10 FIG. 1 is a block diagram for explaining the algorithm of JPEG. The JPEG algorithm is configured with a color space conversion / inverse conversion unit 100, a discrete cosine transform / inverse transform unit 101, a quantization / inverse
15 quantization unit 102, and an entropy coder / decoder unit 103. Generally, non-reversible coding is used so as to gain a high compression ratio. That is, it is impossible to recover all the image data by decompressing the compressed image data. In other
20 words, lossy compression is used more often than lossless compression. Even the lossy compression does not cause any problem in practice. The JPEG algorithm contributes to reducing memory capacity required for compression and decompression of image data and for
25 storage of the compressed image data, and to reducing

time required for exchanging the compressed image data. These advantages have positioned JPEG as the most widely used algorithm for compressing / decompressing still images.

5 Although JPEG is one of the most successful conventional image compression techniques, JPEG still has problems that become apparent as users demand higher and higher resolution. In the case of high resolution still images, block noise and mosquito
10 noise, for example, are apparent. The degradation of image quality is not ignorable.

 The image quality of low bit rate, that is, high compression rate, needs to be improved to solve such problems. JPEG 2000, expected to be a solution,
15 has been created by intensive studies of the compression algorithm requirements of the next generation. JPEG 2000 is expected to be used in parallel with JPEG.

 FIG. 2 is a block diagram for explaining
20 the algorithm of JPEG 2000. The algorithm of JPEG 2000 is configured with a color space converting / inverse converting unit 110, a 2D wavelet transform / inverse transform unit 111, a quantization /
inverse quantization unit 112, an entropy coder /
25 decoder unit 113, and a tag processing unit 114.

As is apparent from the comparison of FIGs. 1 and 2, the JPEG algorithm and the JPEG 2000 algorithm are different in the transforming method. JPEG uses Discrete Cosine Transform (DCT), and JPEG 5 2000 uses Discrete Wavelet Transform (DWT). JPEG 2000 uses DWT because, compared with DCT, DWT provides images of higher quality.

Another difference between JPEG and JPEG 2000 is the tag processing unit 114 provided only in 10 the final stage of JPEG 2000. The tag processing unit 114 creates and interprets codestreams.

The codestreams realize various useful functions of JPEG 2000. FIGs. 3A - 3D are schematic diagrams illustrating sub-bands of each decomposition 15 level in the case that the decomposition level is three. It is possible to compress and decompress the tile-components of a still image up to any octave decomposition level.

As shown in FIGs. 1 and 2, the color space 20 converting /inverse converting units 100 and 110 are often provided at the first stage of the JPEG and JPEG 2000 algorithms, respectively. The color space converting /inverse converting units 100 and 110 convert image data represented in Red, Green, and 25 Blue (RGB) color space or Yellow, Magenta, and Cyan

(YMC) color space into image data represented in YCrCb color space or YUV color space.

The algorithm of JPEG 2000 is described below. Technical terms related to JPEG 2000 are
5 defined in JPEG 2000 Part 1, Final Draft International Standard (FDIS). Some representative technical terms are defined as follows:

1. Code-block: A rectangular grouping of coefficients from the same sub-band of a tile-
10 component.

2. Decomposition level: A collection of wavelet sub-bands where each coefficient has the same spatial impact or span with respect to the source component samples. These include the HL, LH, and HH
15 sub-bands of the same 2-dimensional sub-band decomposition. For the last decomposition level the LL sub-band is also included.

3. Precinct: A rectangular sub-division of a transformed tile-component, within each resolution
20 level, used for limiting the size of packets.

4. Layer: A collection of compressed image data from coding passes of one, or more, code-blocks of a tile component. Layers have an order for encoding and decoding that must be preserved.

25 5. Region of interest (ROI): A collection

of coefficients that are considered of particular relevance by some user-defined measure.

FIG. 4 is a schematic drawing illustrating tile-divided components of a color image. As shown in
5 FIG. 4, components 130, 131, and 132 (represented using RGB color space, in this case) of the color image are divided into a plurality of square regions (tiles) 130t, 131t, and 132t, respectively. The compression and decompression processes are performed
10 by tiles: R00, ..., R15, G00, ..., G15, B00, ... B15. Accordingly, the compression and decompression are independently performed for each component and each tile.

When the color image is compressed, the
15 image data of each tile of each component of the color image is input to the color space controlling unit 110 shown in FIG. 2. After being converted into another color space, the image data are transformed with 2D wavelet transform by the 2D wavelet transform
20 into a frequency domain.

As described above, FIG. 3 shows sub-bands of each decomposition level in the case that the decomposition level is three. The 2D wavelet transform is applied to a tile original image
25 (reference numeral 120) (0LL) (the decomposition

level is zero) obtained by tile-dividing the source image, and the tile source image is separated into sub-bands 1LL, 1HL, 1LH, and 1HH of the decomposition level 1 (reference numeral 121). Then, the 2D wavelet transform is further applied to the low frequency component 1LL of this level, and sub-bands 2LL, 2HL, 2LH, and 2HH of the decomposition level 2 (reference numeral 122) are separated. Likewise, the 2D wavelet transform is applied to the low frequency component 2LL, and sub-bands 3LL, 3HL, 3LH, and 3HH of the decomposition level 3 are separated (reference numeral 123).

In addition, the sub-bands of each decomposition level that are to be encoded are shaded in FIG. 3. The sub-bands 3HL, 3LH, 3HH, 2HL, 2LH, 2HH, 1HL, 1LH, and 1HH are to be encoded and the sub-band 3LL is not encoded in the case that the decomposition level is 3 shown in FIG. 3.

Bits are encoded in a designated order. The quantization unit 112 shown in FIG. 2 generates contexts based on bits adjacent to the bit to be encoded. The quantized wavelet coefficients are divided into non-overlapping rectangular portions called "precincts" within a sub-band. "Precinct" is introduced in order to efficiently use memory space

when JPEG 2000 is implemented.

FIG. 5 is a schematic diagram for explaining the relationship between a precinct and a code-block. The source image 140 is divided into four tiles 140t0, 140t1, 140t2, and 140t3 in the decomposition level 1. As shown in FIG. 5, a precinct 140p4, for example, is made of three rectangle regions that spatially correspond to each other. A precinct 140p6 is made in the same manner. The precincts are numbered from 0 to 8 in raster order. Each precinct is divided into non-overlapping rectangular blocks called code-blocks. In this case, the precinct 140p4 is divided into twelve code-blocks 0-11. A code-block 140b1 indicates a code-block No. 1. A code-block is a unit for entropy coding.

Coefficients obtained by wavelet transformation may be quantized as is into codes. The coefficients may be divided into "bit planes" so as to increase the efficiency of coding. The bit planes are ordered by pixels or code-blocks.

FIG. 6 is a schematic diagram for explaining the process of the ordering. FIG. 6 shows the case in which the source image (32x32 pixels) is divided into four tiles (16x16 pixels), and the size of a precinct and a code-block of the decomposition

level 1 is 8x8 pixels and 4x4 pixels, respectively.
The precincts and the code-blocks are numbered in
raster order.

When pixels need to be extended to the
5 exterior of a tile, a mirroring method is used. The
wavelet transformation is performed with a reversible
(5, 3) filter. The wavelet coefficients of the
decomposition level 1 are determined as described
above.

10 A representative "layer" of tile 0,
precinct 3, code-block 3 is also shown in FIG. 6. One
may easily understand the structure of a layer by
looking at the wavelet coefficients in the direction
of the bit plane. A layer is made of one or more bit
15 planes. In this case, the layers 0, 1, 2, 3 are made
of three bit planes 1, 3, 1. The closer to LSB is a
bit plane included in a layer, the earlier the layer
is quantized. To the contrary, the closer to MSB is a
bit plane included in a layer, the later the layer is
20 quantized. Discarding layers that are closer to LSB
in the order of closeness is called truncation. The
quantizing rate can be finely adjusted with
truncation.

The entropy encoding unit 113 shown in FIG.
25 2 encodes tiles of each component by stochastic

estimation based on context and bits to be encoded.
All components of the source image are encoded by
tiles. The tag processing unit 114 combines all
encoded data from the entropy encoding unit 113 into
5 a codestream, and attaches a tag to the codestream.

FIG. 7 is a data diagram showing the
structure of a codestream. Tag information called a
main header 150 and tile-part headers 151 are
attached to the top of the codestream and the top of
10 each tile-part structuring each tile, respectively.
The encoded data (bit stream 152) of each tile follow
the tile-part header 151. A tag 153 (End of
codestream) is attached to the end of the codestream.

Contrary to the encoding process, image
15 data are reproduced from the codestream of each tile
of each component in the decoding process. The
decoding process is described below with reference to
FIG. 2.

The tag processing unit 114 interprets tag
20 information attached to the input codestream, and
divides the input codestream into codestreams of each
tile of each component. Decoding is performed by a
codestream of each tile of each component. The
position of an object bit is determined in the order
25 based on the tag information provided to the

codestream. The inverse quantizing unit 112 generates context based on bits (that are already decoded) adjacent to the object bit. The entropy decoding unit 113 performs decoding based on the context and the
5 codestream by stochastic estimation. The object bit is decoded accordingly and stored at the position.

As described above, the decoded data are spatially divided by frequency range. The image data of each tile of each component are restored by
10 performing two-dimensional wavelet inverse transformation by the 2D wavelet inverse transforming unit 111. The restored data are converted into data represented by their source color space by the color space inverse converting unit 110.

15 Inverse processing of conventional JPEG codes is basically the same as inverse processing of JPEG 2000 codes, but is different only in that 8x8 pixel rectangular blocks instead of the tiles are transformed by JPEG two-dimensional discrete cosine
20 transformation.

General processing of still images is described above. JPEG and JPEG 2000 also become applicable to motion pictures. A motion picture can be structured with a plurality of frames, each being
25 a still picture, that are framed at an appropriate

frame speed. The motion picture can be reproduced by displaying the plurality of still pictures at the frame speed. This technique is called motion compression /decompression processing of still pictures. Compared with MPEG, which is widely used for motion pictures, this technique has an advantage in that each frame is a still picture of high quality. Accordingly, the motion compression /decompression processing of still pictures is drawing the attention of business users such as broadcasters. It is probable that this technique will pervade to consumers.

[STRUCTURE OF IMAGE FORMING APPARATUS]

A preferred embodiment of the present invention is described below.

FIG. 8 is a cross-section of an image forming apparatus according to an embodiment of the present invention. The image forming apparatus 1 is provided with an image reading unit 2 and an image forming unit 21 that forms images on a recording medium such as a sheet of paper based on image data provided by the image reading unit 2. The image reading unit 2 is a scanner, for example, and the image forming unit 21 is a printer, for example.

A contact glass 3 is provided on the top of

the chassis of the image reading unit 2. Although not shown, a document is placed on the contact glass 3 and read. The document is placed on the contact glass 3 with its face down. A platen cover 4 is provided
5 above the contact glass 3 and retains the document..

Below the contact glass 3, the following are provided: a first moving unit 7 in which a light source 5 and a mirror 6 are provided, a second moving unit 10 in which two mirrors 8 and 9 are provided,
10 and an optical system 13 in which a conversion lens 11 and a charge coupled device (CCD) image sensor 12 are provided. The light radiated by the light source 5 is reflected by the document. The reflective light is led to the conversion lens 11 through the mirrors
15 6, 8, and 9. The converged reflective light is detected by the CCD image sensor 12. The CCD image sensor 12 is an optoelectronic device for converting the converged reflective light into optoelectronic conversion data. The optoelectronic conversion data
20 are represented by voltages depending on the intensity of the reflective light.

The first and second moving units 7 and 10 can move back and forth along the contact glass 3. In a reading operation (described below), the first and
25 second moving units 7 and 10 move at a speed ratio of

2:1. As the first and second moving units 7 and 10 move along the contact glass 3, the document is scanned by the optical system 13.

The image forming unit 21 is provided with
5 a medium path 26 along which a medium container 22, an electrophotography type printer engine 23, a fixing unit 24, and a discharge unit 25 are provided.

The printer engine 23 is provided with a charge unit 27, an exposure unit 28, a development
10 unit 29, a transfer unit 30, a cleaner unit 31, and a photosensitive body 32. The printer engine 23 forms a toner image on the photosensitive body 32 by electrophotography, transfers the formed toner image to the recording medium, and fixes the transferred
15 toner image with the fixing unit 24. The printer engine 23 forms images by electrophotography in this case. However, the printer engine 23 may use another image forming technique such as an ink jet system, a sublimation type thermal transfer system, and a
20 direct thermal system.

The image forming apparatus 1 is controlled by a controller including a plurality of microprocessors. FIG. 9 is a block diagram showing the structure of the controller. The controller is
25 configured by a central processing unit (CPU) 41, a

read only memory (ROM) 42, a random access memory (RAM) 43, an image processing unit (IPU) 45, and an input/output port (I/O) 46 connected via a bus 44. The CPU 41 performs various computations and
5 centrally controls processing such as image processing. The ROM 42 stores therein various programs and fixed data that are used by the CPU 41. The RAM 43 provides the CPU 41 a working memory region. The IPU 45 is provided with various hardware
10 devices for processing images.

The ROM 42 is provided with a non-volatile memory such as a flash memory. The program stored in the ROM 42 can be replaced with a program to be downloaded from an external source (not shown)
15 through the I/O port 46. The CPU 41 also controls such a replacement.

FIG. 10 is a functional block diagram showing an image processing apparatus 51. The image processing apparatus 51 includes an image compression
20 encoding unit 52 and an image composite unit 53. The image compression encoding unit 52 realizes an image compression coding unit. The image compression encoding unit 52 converts an image into a codestream using the JPEG 2000 algorithm. (The image is read by
25 the image reading apparatus 2, and may be processed

by the IPU 45. The processing by the IPU 45 may include white shading adjustment, for example.) For the conversion, the image compression encoding unit 52 divides the image into one or more rectangular regions (tiles) and transforms pixel values of each rectangular region with the discrete wavelet transformation into compression codes. The encoding is performed hierarchically. The image composite unit 53 composites codestreams of a plurality of images into a composite image.

The image compression coding unit 52 includes the functional blocks described with reference to FIG. 2, and encodes the images into compression codes with the JPEG 2000 algorithm described above. The function of the image compression encoding unit 52 may be performed by hardware included in the IPU 45, or may be performed by the CPU 41 that runs a program stored in the ROM 42. The function of the image composite unit 53 may be performed by hardware included in the IPU 45, or may be performed by the CPU 41 that runs a program stored in the ROM 42.

FIG. 11 is a block diagram showing the functional structure of the image composite unit 53. The image composite unit 53 includes an image reading

unit 54, a header/code data separation unit 55, a header processing unit 56, a code data processing unit 57, a codestream generation unit 58, and a composite setting unit 59.

5 The operation of the image composite unit 53 in which a plurality of component images are composited into one composite image is described. It is assumed that each codestream input to the image composite unit 53 is structured by a tile of the same
10 size. However, those skilled in the art easily understand that the embodiment of the present invention is not limited to this case.

 A user operates an operations panel (not shown) of the image forming apparatus 1 so as to
15 composite a plurality of images into a composite image. The image reading unit 2 reads the plurality of images (of documents) into a plurality of items of digital image data. The digital image data may be processed by the IPU 45 for a white shading
20 adjustment, for example. The digital image data are encoded into a plurality of codestreams, and output to the image composite unit 53.

 The user designates through the operations panel (not shown) the number of images that the user
25 desires to composite into one image (one page), and

the composite setting unit 59 determines the number of images to be arranged in the horizontal and vertical directions. The composite setting unit 59 embodies a composite setting unit. For example, if
5 the user desires to composite four images into one page, the number of images arranged in the horizontal directions and the vertical directions may be 2 x 2, respectively. The image reading unit 54 reads the codestreams to be composited one by one.

10 The first codestream 61 is read, and separated into the header portion and the code data portion by the header/code data separation unit 55. The header processing unit 56 adjusts the image size of the main header to the image size after
15 compositing. The composite setting unit 59 determines the image size after compositing. The header processing unit 56 generates a new tile-part header, and attaches a tile index to the generated tile-part header.

20 Then, the image reading unit 54 reads the second or higher codestream 62, 63, ..., and the header/code data separation unit 55 separates them into headers and code data. The header processing unit 56 converts the main headers into tile-part
25 headers. A tile index is attached to each tile-part

header. The headers of all image data are processed in the same manner.

The code data processing unit 57 embodies a code data processing unit. The code data processing unit 57 extracts only predetermined code data out of each item of code data separated by the header/code data separation unit 55. Specifically, the code data processing unit 57 extracts only wavelet transform coefficients of a specific layer. The code data processing unit 57 outputs the extracted code data to the codestream generation unit 58.

The codestream generation unit 58 embodies a codestream generation unit. The codestream generation unit 58 generates a codestream in compliance with the requirement of JPEG 2000 based on the header generated by the header processing unit 56 and the code data extracted by the code data processing unit 57. As a result, a new codestream 61' representing an image in which the plurality of images are composited is generated. This codestream 61' is used by the printer engine 23. That is, the codestream 61' is stored in an image memory (not shown), and printed by the printer engine 23 after being expanded by an image expanding apparatus.

FIG. 12 is a data diagram for explaining

the structure of a plurality of codestreams before being composited by the image composite unit 53. FIG. 12 shows the following: the main header, the tile-part header, a bit stream, an end of codestream (EOC) marker of the first codestream 61; the main header, the tile-part header, a bit stream, an EOC marker of the second codestream 62;...; and the main header, the tile-part header, a bit stream, an EOC marker of the n-th codestream 6N.

10 FIG. 13 is a data diagram for explaining the structure of the codestream after compositing by the image composite unit 53. FIG. 13 shows the main header, a tile-part header, a bit stream, another tile-part header, another bit stream, yet another tile-part header, yet another bit stream, and the EOC marker of the codestream 61' of the composite image. The codestream 61' of the composite image includes only the wavelet coefficients LL1, HL1, LH1, and HH1 of the decomposition level 1 of each component image.

20 FIG. 14 is a flow diagram for explaining processing of the image forming apparatus 1 for compositing a plurality of images.

 When the user gives a request for photocopying ("Yes" in step S1), a determination is made whether the user gives a request for compositing

25

a plurality of images (step S2). If a negative determination is made ("No" in step S2), the process is terminated.

If a determination is made that the user
5 gives a request for compositing a plurality of images
("Yes" in step S2), a determination is made whether
the image reading unit 54 has read all codestreams
that are to be composited (step S3). If the image
reading unit 54 has already read all codestreams
10 ("Yes" in step S3), the process proceeds to step S11.
If not yet ("No" in step S3), the remaining images
are read (step S4). The header/code data separation
unit 55 separates the codestream into the header
portion and the code data portion (step S5). Step S5
15 embodies the step of separating header/code data.

Then, a determination is made whether the
read codestream is the first codestream (step S6). If
the read codestream is the first codestream ("Yes" in
step S6), the main header and the tile-part header
20 after compositing are generated by the header
processing unit 56 based on the main header and the
tile-part header of the read codestream (step S7). If
the read codestream is not the first codestream ("No"
in step S6), the tile-part header after compositing
25 is generated by the header processing unit 56 based

on the main header and the tile-part header of the
read codestream (step S8). Steps S7 and S8 embody the
step of processing the header. As described above,
the wavelet transform coefficients of a decomposition
5 level are extracted (step S9). Step 9 embodies the
step of processing code data.

Then, a counter for counting the number of
the read codestreams is incremented by one (step S10).
The process returns to step S3. The steps from S3
10 through S10 are repeated until all codestreams to be
composited are processed. When all the codestreams to
be composited are processed ("Yes" in step S3), the
new codestream after compositing is generated (step
S11). Then, the process is terminated. Step S11
15 embodies the step of generating a codestream.

FIG. 15 is a flow chart of the operation of
the image forming apparatus 1 in which a composited
codestream is separated (decomposed) into a
plurality of codestreams. When the user gives an
20 instruction for copying ("Yes" of step S21), the
image forming apparatus 1 further determines whether
the user gives an instruction for separating by using
the operations panel (not shown) (step S22). If the
user does not instruct to separate an image ("No" in
25 step S22), the process ends. If the user instructs to

separate an image ("Yes" in step S22), the image reading unit 54 reads the codestream of the composite image (step S23). The header / code data separation unit 55 separates the read codestream (step S24).

5 Step S24 realizes a step of separating header and code data.

A determination is made whether all separations have been completed (step S25). If a determination is made that all separations have been
10 completed ("Yes" in step S25), the process ends. If a determination is made that all separations have not been completed ("No" in step S25), the header processing unit 56 edits the separated header. The codestream generating unit 57 generates a new
15 codestream by adding a new header to the separated code data. Accordingly, the image is separated into a plurality of images (step S26). Step S26 realizes a step of generating a new codestream. Then, a counter for counting the number of separated images is
20 incremented (step S27). The process returns to step S25. Steps S25 through S27 are repeated until all separations are completed. When all separations have been completed, the process ends.

As described above, if there is a composite
25 image with four images (two in the width directions

and two in the height directions) composited therein,
the user can instruct the image forming apparatus to
separate the composite image into four separated
component images by using the operations panel (not
5 shown).

In response to the instruction from the
user, the scanner 2 reads the composite image. The
image compression coder unit 52 generates a
codestream as shown in FIG. 13 in which the scanned
10 composite image is divided into four (2x2) images.
The codestream is separated into headers and code
data (step S24), and four codestreams, such as shown
in FIG. 12, in which each item of code data is a tile,
are formed (step S26). Each codestream is printable
15 by the printer 21. Accordingly, the composite image
is separated into four printed component images.

As described above, the image processing
apparatus 51 can composite a plurality of images into
a composite image and separate the composite image
20 into a plurality of images. However, the image
processing apparatus 51 may be structured to process
only one of the two functions.

In the above description, it is assumed
that each codestream to be composited is formed by a
25 single tile in order to make the description easy.

Even if each codestream to be composited is formed by a plurality of tiles, images can be composited in almost the same manner as described above except that there is a minor difference in attaching tile indexes.

5 It is also assumed that each image composited in the composite image is formed by a single tile in order to make the description simple. Each image composited in the composite image may be formed by a plurality of tiles. In this case, a
10 codestream is generated for each group of tiles that forms an image.

 According to the above embodiment, the image processing apparatus 51 is built into the image forming apparatus 1. According to another embodiment,
15 the image processing apparatus 51 may be applied to various electronics apparatuses. For example, the image forming apparatus 51 may be embodied by an application program running on an information processing apparatus such as a personal computer. In
20 this case, the image processing apparatus can composite various image data. The application program may be provided stored in an optical disk, an optical-magnetic disk, and a flexible disk, for example. According to another embodiment, the
25 application program may be downloaded via a network

such as the Internet.

The preferred embodiments of the present invention are described above. The present invention is not limited to these embodiments, but various
5 variations and modifications may be made without departing from the scope of the present invention.

This patent application is based on Japanese Priority Patent Applications No. 2002-244877 filed on August 26, 2002, and No. 2003-001227 filed
10 on January 7, 2003, the entire contents of which are hereby incorporated by reference.